Modelli

deve essere lineare (No variabili logiche moltiplicate o funzioni di max() e min())

- Trovare una funzione obbiettivo (max, min)
- Aggiungere variabili quantitative o logiche
- Trovare vincoli in base al problema
  - vincoli di copertura: quando $\sum x_i \geq 1$ ovvero ci assicuriamo che ogni "elemento" sia associato ad almeno una "destinazione". Se al posto di "≥" metto "=" diventa un problema di partizione
  - vincoli di interezza: affermano che le variabili in tali vincoli sono intere ($\in \mathbb{Z}$)
  - vincoli di soglia: permettono di lavorare anche con i valori assoluti e variabili ausiliarie (approssimazioni superiori o inferiori del valore cercato).
  - vincoli di semiassegnamento: ciascun "oggetto" viene assegnato ad uno e uno solo "elemento" (NON è suriettiva).

Grafi (Modelli)

- Trovare la sorgente (s) e il pozzo (t) (possono essere molteplici)
- Trasferire il flusso da s a t considerando c_{ij} (costo attivazione arco) b_i (bilanci dei nodi).
- Trovare una funzione obbiettivo (max, min)
- Trovare i vincoli relativi al problema
  - Vincoli di conservazione del flusso: $\sum_{j \in BN(i)} x_{ij} - \sum_{j \in FN(i)} x_{ij} = b_i \quad (\sum_{i \in N} b_i = 0)$
  - Vincoli di capacità: $0 \leq x_{ij} \leq u_{ij} \quad (i,j) \in A$
- Se immaginiamo il flusso da mandare come dei pacchetti allora s dovrà inviare esattamente P pacchetti quindi come bilancio avrà $-P$ mentre t dovrà ricevere P pacchetti quindi come bilancio avrà P.

$$\hookrightarrow \quad b_i = \begin{cases} -P & i = s \\ 0 & i \neq s,t \\ P & i = t \end{cases}$$

Grafi:

- Costruzione di un albero dei cammini minimi (contiene il cammino di costo minimo da s a t). Per verificare che un albero sia dei cammini minimi bisogna controllare le condizioni di Bellmann ($d_i + c_{ij} \geq d_j$) per ogni $(i,j) \in A$. Ovviamente $d_r = 0$ dove r é la radice e d_i é il costo del cammino da r ad i.
  Con l'aggiunta di un arco fittizio di costo $M = (n-1)C_{max} + 1$ possiamo trovare un albero dei cammini minimi.
  Il costo di un albero dei cammini minimi é la somma dei bilanci dei suoi nodi.
- Per "visitare" un grafo possiamo usare Dijkstra se tutti gli archi hanno costo ≥ 0 e Bellmann-Ford se alcuni sono < 0.
- Costruzione di un albero di copertura di costo minimo (sottografo connesso privo di archi). Esistono due metodi per verificare che un albero sia di copertura di costo minimo:
  - Ottimalitá per tagli: (N, A_T) é un albero di copertura di costo minimo <=> il costo di ciascun $(i,j) \in A_T$ é $\leq$ del costo di ciascun arco del taglio che si viene a formare rimuovendo (i,j).
  - Ottimalitá per cicli: (N, A_T) é un albero di copertura di costo minimo <=> il costo di ciascun $(i,j) \notin A_T$ é $\geq$ al costo di ciascun arco del ciclo che si viene a formare aggiungendo (i,j).
- Taglio: Partizione di un grafo in 2 insiemi (N', N'') t.c. $N', N'' \subseteq N$, $N' \cap N'' = \emptyset$, $N' \cup N'' = N$.
- Algoritmi di inserzione e cancellazione di archi:
  - Algoritmo di Kruskal: Si ordinano gli archi per costo non decrescente e vengono esaminati con tale ordine. Se l'arco analizzato non forma un ciclo con gli archi giá inseriti lo si aggiunge, altrimenti lo si scarta.
  - Algoritmo di Prim: Fa esclusivamente inserzioni. Preso un taglio che esclude un nodo i dagli altri, scegliamo l'arco di costo minimo tra gli

archi del taglio inserendolo, continuo l'algoritmo creando un nuovo taglio che separa il nodo i ed il nodo j (che abbiamo collegato precedentemente) dal resto del grafo e proseguiamo scegliendo l'arco di costo minimo del taglio che si crea, e così via.

Grafi (Problemi di flusso massimo)

- Ogni nodo ha un bilancio b_i, mentre ogni arco ha un flusso già presente x_{ij} ed una capacità u_{ij}
- Se $x_{ij} = u_{ij}$ l'arco si dice **saturo**, mentre $u_{ij} - x_{ij}$ è la **capacità residua**
- In un grafo è possibile trovare un **cammino aumentante** in grado di aumentare il valore del flusso di una quantità $\theta = \min\{u_{ij} - x_{ij} : (i,j) \in P\}$ dove "P" è il cammino.
>0 Gli archi che "seguono" il verso del cammino vengono detti **concordi** mentre quelli che vanno nella direzione opposta sono detti **discordi**.
- La costruzione del **grafo residuo** avviene partendo da un grafo e inserendo nel grafo residuo gli archi non saturi nello stesso verso del grafo originale e gli archi non vuoti (che contengono flusso) in verso opposto e non inserendo gli archi saturi dato che da essi non può circolare ulteriore flusso.
- (N_s, N_T) è un **taglio che separa s da t** se $s \in N_s$ e $t \in N_t$ indichiamo con $u(N_s, N_t) = \sum_{(i,j) \in A_+} u_{ij}$ la **capacità del Taglio** e con $x(N_s, N_t) = \sum_{(i,j) \in A^+} x_{ij} - \sum_{(i,j) \in A^-} x_{ij}$ il **flusso nel taglio**.
- Il massimo valore del flusso ammissibile è uguale alla minima capacità dei tagli che separano s da t

Grafi (Problemi di flusso di costo minimo)

- Ogni arco avrà un costo c_{ij} e una capacità u_{ij}, con l'utilizzo di uno pseudoflusso (grafo dove ogni arco ha un flusso x_{ij}) possiamo costruire un grafo residuo con lo stesso procedimento dei pb. di flusso di costo massimo ma se $x_{ij} < u_{ij}$ nell'arco metteremo $u_{ij} - x_{ij}$ e costo c_{ij}, se $x_{ij} > u_{ij}$ metteremo x_{ij} e costo $-c_{ij}$.
- A partire da uno pseudoflusso x è possibile calcolare gli sbilanciamenti dei nodi, un nodo ha surplus se trattiene del flusso che deve inviare e ha deficit se non riceve del flusso che dovrebbe ricevere. Indichiamo con $e_i(x) = \sum x_{ji} - \sum x_{ij} - b_i$ lo sbilanciamento del nodo i a partire dallo pseudoflusso x e se $e_i(x) = 0$ il nodo si dice bilanciato, se $e_i(x) > 0$ il nodo ha surplus, $e_i(x) < 0$ vuol dire che il nodo ha deficit.